THE DEFINITION OF THE LL(1) PARSING TABLE — NULLABLE

Algorithmic definition: the function NULLABLE

Let $\mathcal{G} = (\mathbf{N}, \mathbf{T}, \rightarrow, S)$ be a context-free grammar. $\mathtt{NULLABLE}_i$ are functions on on $\mathcal{G}$ defined as follows

1. $\mathtt{NULLABLE}_0(\mathcal{G}) = \{ A \mid A \rightarrow \varepsilon \text{ in } \mathcal{G}\}$
2. $\mathtt{NULLABLE}_{i+1}(\mathcal{G}) = \mathtt{NULLABLE}_i(\mathcal{G})$

$$\cup \{ A \mid A \rightarrow A_1 \dots A_n \text{ in } \mathcal{G} \ \wedge \ A_1, \dots, A_n \in \mathtt{NULLABLE}_i(\mathcal{G}) \}$$

---

* it is easy to show that $\mathtt{NULLABLE}_i(\mathcal{G}) \subseteq \mathtt{NULLABLE}_{i+1}(\mathcal{G}) \subseteq \mathbf{N}$
* therefore there is k such that $\mathtt{NULLABLE}_k(\mathcal{G}) = \mathtt{NULLABLE}_{k+1}(\mathcal{G})$

then $\mathtt{NULLABLE}(\mathcal{G}) = \mathtt{NULLABLE}_k(\mathcal{G})$

DEFINITION OF LL(1) PARSING TABLES: FIRST

Algorithmic definition: the function FIRST

Let $\mathcal{G} = (\mathbf{N}, \mathbf{T}, \rightarrow, S)$ be a context-free grammar. $\mathtt{FIRST}_i$ are functions on $\mathbf{N} \cup \mathbf{T}$ and defined as follows

1. $\mathtt{FIRST}_i(\mathtt{t}) = \{ \mathtt{t} \}$, with $\mathtt{t} \in \mathbf{T}$ `// for every i`

2. $$\mathtt{FIRST}_0(A) = \begin{cases} \{ \varepsilon \} & \text{if } A \in \mathtt{NULLABLE}(\mathcal{G}) \\ \emptyset & \text{if } A \notin \mathtt{NULLABLE}(\mathcal{G}) \wedge A \in \mathbf{N} \end{cases}$$

3. $$\mathtt{FIRST}_{i+1}(A) = \mathtt{FIRST}_i(A) \cup \bigcup_{\substack{A \rightarrow \alpha_1 \cdots \alpha_n \text{ in } \mathcal{G} \\ \forall\, \mathtt{i} \in \mathtt{1..k-1} \,:\, \alpha_i \in \mathtt{NULLABLE}(\mathcal{G})}} \mathtt{FIRST}(\alpha_k) \setminus \{ \varepsilon \}$$

∗ it is easy to show that, for every $\mathtt{i}$: $\mathtt{FIRST}_i(A) \subseteq \mathtt{FIRST}_{i+1}(A) \subseteq \mathbf{T} \cup \{ \varepsilon \}$

∗ therefore there is $\mathtt{k}$ such that, for every A, $\mathtt{FIRST}_k(A) = \mathtt{FIRST}_{k+1}(A)$

then $\mathtt{FIRST}(A) = \mathtt{FIRST}_k(A)$

DEFINITION OF `LL(1)` PARSING TABLES: `FOLLOW`

Algorithmic definition: the function `FOLLOW`

Let $\mathcal{G}$ = (**N**, **T**, →, S) be a context-free grammar. $\mathtt{FOLLOW}_i$ are functions on **N** and defined as follows

1. $\mathtt{FOLLOW}_0(\mathtt{S}) = \{ \$ \}$ and $\mathtt{FOLLOW}_0(\mathtt{A}) = \varnothing$

2. $\mathtt{FOLLOW}_{i+1}(\mathtt{X}) = \mathtt{FOLLOW}_i(\mathtt{X}) \bigcup_{\mathtt{Z} \to \delta\, \mathtt{X}\, \gamma \text{ in } \mathcal{G}} \mathtt{FIRST}(\gamma)\setminus\{\varepsilon\}$
$\bigcup_{\mathtt{Z} \to \delta\, \mathtt{X}\, \gamma \text{ in } \mathcal{G} \text{ and } \mathtt{NULLABLE}(\gamma)} \mathtt{FOLLOW}_i(\mathtt{Z})$

---

* it is easy to show that, for every `i`: $\mathtt{FOLLOW}_i(\mathsf{A}) \subseteq \mathtt{FOLLOW}_{i+1}(\mathsf{A}) \subseteq \mathbf{T} \cup \{ \$ \}$
* therefore there is `k` such that, for every A, $\mathtt{FOLLOW}_k(\mathsf{A}) = \mathtt{FOLLOW}_{k+1}(\mathsf{A})$

then $\mathtt{FOLLOW}(\mathsf{A}) = \mathtt{FOLLOW}_k(\mathsf{A})$

remarks: (1) when the initial symbol does not appear on the rhs of productions, "`$`" is the unique symbol in its `FOLLOW`

`(2)` `FOLLOW` never contains "ε"